# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*



**PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

| 1. REPORT DATE *(DD-MM-YYYY)* | 2. REPORT TYPE | 3. DATES COVERED *(From - To)* |
|---|---|---|
| 01-04-2009 | Journal Article | March 2006 - March 2007 |

**4. TITLE AND SUBTITLE**

Multi-Target/Multi-Sensor Tracking using Only Range and Doppler Measurements

**5a. CONTRACT NUMBER**
N/A

**5b. GRANT NUMBER**
N/A

**5c. PROGRAM ELEMENT NUMBER**
612303

**6. AUTHOR(S)**
Ross Deming*, John Schindler*, Leonid Perlovsky

**5d. PROJECT NUMBER**
2311

**5e. TASK NUMBER**
HE

**5f. WORK UNIT NUMBER**
01

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**

Electromagnetic Scattering Branch
Air Force Research Laboratory, Sensors Directorate
80 Scott Drive
Hanscom AFB, MA 01731-2909

*Research Consultant
GDIT
Fairfax, VA 22030

**8. PERFORMING ORGANIZATION REPORT NUMBER**
N/A

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

Electromagnetics Technology Division SC: 437490
Sensors Directorate
Air Force Research Laboratory
80 Scott Drive Hanscom AFB, MA 01731-2909

**10. SPONSOR/MONITOR'S ACRONYM(S)**
AFRL-RY-HS

**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**
AFRL-RY-HS-TP-2009-0008

**12. DISTRIBUTION/AVAILABILITY STATEMENT**

Distribution A: Approved for UNLIMITED DISTRIBUTION

**13. SUPPLEMENTARY NOTES**

Research support by Air Force Office of Scientific Research. Cleared for public release by ESC Public Affairs Office Clearance # 06-1168; Published in IEEE Transactions on Aerospace & Electronic Systems, Vol. 45, No. 2, April 2009

**14. ABSTRACT**

A new approach is described for combining range and Doppler data from multiple radar platforms to perform multi-target detection and tracking. In particular, azimuthal measurements are assumed to be either coarse or unavailable, so that multiple sensors are required to triangulate target tracks using range and Doppler measurements only. Increasing the number of sensors can cause data association by conventional means to become impractical due to combinatorial complexity, i.e., an exponential increase in the number of mappings between signatures and target models. In the new approach, the data association is performed probabilistically, using a variation of expectation-maximization (EM). Combinatorial complexity is avoided by performing an efficient optimization in the space of all target tracks and mappings between tracks and data. The full, multi-sensor, version of the algorithm is tested on simulated data. The results demonstrate that accurate tracks can be estimated by exploiting spatial diversity in the sensor locations. These results are promising, and demonstrate robustness in the presence of nonhomogeneous clutter.

**15. SUBJECT TERMS**

Complex-point dipoles, cylindrical scanning, near-field measurements, probe correction, spherical scanning.

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | Leonid I. Perlovsky |
| U | U | U | UU | 20 | 19b. TELEPHONE NUMBER *(Include area code)* 781-377-1728 |

Reset

**Standard Form 298** (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

# Multi-Target/Multi-Sensor Tracking using Only Range and Doppler Measurements

**ROSS DEMING,** Senior Member, IEEE
General Dynamics Information Technology, Inc.

**JOHN SCHINDLER,** Life Fellow, IEEE
General Dynamics Information Technology, Inc.

**LEONID PERLOVSKY,** Senior Member, IEEE
Air Force Research Laboratory

**A new approach is described for combining range and Doppler data from multiple radar platforms to perform multi-target detection and tracking. In particular, azimuthal measurements are assumed to be either coarse or unavailable, so that multiple sensors are required to triangulate target tracks using range and Doppler measurements only. Increasing the number of sensors can cause data association by conventional means to become impractical due to combinatorial complexity, i.e., an exponential increase in the number of mappings between signatures and target models. When the azimuthal resolution is coarse, this problem will be exacerbated by the resulting overlap between signatures from multiple targets and clutter. In the new approach, the data association is performed probabilistically, using a variation of expectation-maximization (EM). Combinatorial complexity is avoided by performing an efficient optimization in the space of all target tracks and mappings between tracks and data. The full, multi-sensor, version of the algorithm is tested on simulated data. The results demonstrate that accurate tracks can be estimated by exploiting spatial diversity in the sensor locations. Also, as a proof-of-concept, a simplified, single-sensor range-only version of the algorithm is tested on experimental radar data acquired with a stretch radar receiver. These results are promising, and demonstrate robustness in the presence of nonhomogeneous clutter.**

Manuscript received February 7, 2007; revised October 11, 2007; released for publication February 28, 2008.

IEEE Log No. T-AES/45/2/933007.

Refereeing of this contribution was handled by E. Chornoboy.

All work was conducted on-site at the Air Force Research Laboratory, Hanscom Air Force Base.

An abbreviated version of this work was published in *Proceedings of the IEEE Radar Conference*, 2007, and the *IEEE International Conference on Integration of Knowledge Intensive Multi-Agent Systems*, 2007.

Authors' address: Air Force Research Laboratory, Sensors Directorate/RYHE, 80 Scott Drive, Hanscom, AFB, MA 01731, E-mail: (ross.deming@hanscom.af.mil; jkschindler@verizon.net; leonid.perlovsky@hanscom.af.mil).



## I. INTRODUCTION

We present a new approach for multi-target detection and tracking, in which information from multiple, spatially diverse, radar sensors is combined to improve track reliability and accuracy. In particular, we treat the difficult case in which azimuthal measurements are either coarse or unavailable, so that multiple sensors are required to triangulate target tracks using range and Doppler measurements only. However, data association in this problem is extremely complex for several reasons. First, the coarse azimuthal resolution can result in significant overlap between the signatures of the multiple targets and clutter. Second, increasing the number of sensor platforms leads to an exponential increase in the number of mappings between signatures and hypothesized targets. Thus, it becomes impractical to sort out the associations using combinatorial approaches.

Multiple hypothesis tracking (MHT) [1, 2], a benchmark multi-target tracking algorithm, performs data association using an exhaustive evaluation of all mappings between targets and data samples, and is therefore subject to a combinatorial explosion as the amount of data and the number of sensors increase. Pruning or gating are typically used to alleviate the computational burden by eliminating the less likely hypotheses, however valid hypotheses may also be eliminated in the process. Therefore, data having low signal-to-clutter ratio (S/C) may be discarded, which can lead to missed detections. An alternative to MHT is joint probabilistic data association (JPDA) [3–5] which is more efficient than MHT because one only needs to evaluate the association probabilities separately at each time step. Whereas more simplistic single scan methods (such as nearest neighbor approaches) consider only the single observation closest to the predicted state, JPDA is more robust since the state of each track is updated using a weighted average of all measurements falling within its validation region at the current time step. However, since it is a single scan process, not all possible data-to-track mappings are considered. A drawback of JPDA is that while it is appropriate for track maintenance, it lacks an explicit mechanism for track initiation [6]. Both MHT and JPDA have been adapted for multi-sensor scenarios [5].

A data association approach based upon linear programming has been proposed which, like JPDA, updates the track states using weighted averages of measurements [6]. Results from computer simulations indicate significantly lower computational complexity than JPDA, as well as improved accuracy. Moreover, this method provides an explicit mechanism for track initiation.

Recently, sequential Monte Carlo methods, a.k.a. "particle filters," have been adapted for multi-target

tracking problems [7–9]. Whereas Kalman filters are used in traditional JPDA methods for updating track states, particle filters are more appropriate for situations involving nonlinear state and measurement equations and non-Gaussian noise. Data association for particle filter methods can be performed in various ways, as described in [8].

Both MHT and JPDA assume—often correctly—that a target can generate at most one measurement per scan. However, if this constraint is relaxed, data association can be formulated using a continuous optimization procedure, notably expectation-maximization (EM) [10–12], rather than combinatorics. Thus, an efficient "hill climbing" optimization is performed in the space of all model parameters and all possible mappings between data samples and hypothesized targets. An important advantage of this formulation is that computational complexity scales only linearly with the number of targets and sensors, whereas combinatorial approaches such as MHT scale exponentially. Thus, trackers based upon the EM formulation would, in principle, be practical for scenarios with high clutter and/or high densities of targets. Streit, et al. [13–15] developed the probabilistic multihypothesis tracker (PMHT) that utilizes EM to perform data association while simultaneously estimating tracks based upon multiple scans of data. This approach has also been extended by other authors, for example to cases with multiple sensors [16–18] and maneuvering targets [19]. Avitzour [20] and Perlovsky [21, 22] also, independently, developed maximum-likelihood procedures for multi-target tracking which use EM for data association. Perlovsky demonstrated mathematically [22, 23], using Cramer-Rao bound analysis, that the utilization of classification features within the tracker is equivalent to an improvement in S/C ratio in high-clutter tracking environments. Subsequently, he developed a version of the algorithm in which classification and tracking are performed concurrently [22, 24–26]. Here, if they are available, classification features (e.g., radar cross section (RCS), length, etc.) are placed on an equal footing with tracking features (e.g., range, Doppler, bearing, etc.). The model is then a mixture of different types of target and clutter components in the combined space of tracking and classification features. Like PMHT, Perlovsky's approach is a multi-scan (i.e., "batch") algorithm.

Perlovsky's approach differs from PMHT chiefly in the choice of track model. In PMHT the motion of each target is modeled using a set of discrete-time state transition equations, and therefore a Kalman smoother is used to estimate track parameters in the M-step of the EM iterations [13–15] (in nonlinear/non-Gaussian cases the target states are obtained using dynamic programming rather than Kalman smoothers). In contrast, Perlovsky's approach is flexible with respect to the choice of track model, but normally the choice is to include continuous polynomial models (e.g., constant velocity, constant acceleration, etc.), or piecewise polynomial models, for target trajectories. The use of polynomial models leads to very simple parameter update formulas in the M-step, for example, simple matrix inversions which are similar in structure to polynomial regression [22]. Of course, under certain conditions the discrete-time state transition equations used in PMHT would be equivalent to polynomial models. Another difference is in the choice of optimization criterion. Whereas in PMHT the goal is maximization of the a posteriori probability (MAP) [15], Perlovsky uses maximum likelihood estimation (MLE) in the case of sampled data, and minimization of the cross-entropy in the case of pixelated data [22]. Avitzour's procedure is maximum likelihood, like Perlovsky's, however different tracking models are specified due to differences in the particular applications.

The algorithm described in the present paper is developed along the lines of Perlovsky's general approach, and utilizes EM for data association. However, a major new aspect considered here is the data model, which is particularly complicated, and arises from combining data from multiple sensors, where azimuthal measurements are absent. Thus, the goal here is to use multiple sensors to triangulate target tracks using range and Doppler measurements only, while performing data association probabilistically. An abbreviated version of this work was published previously [30, 31], although some results and most of the mathematical details were not included.

An important consideration for any tracking algorithm is computational cost. When applied to a benchmark (single-sensor, single-target) tracking problem, it was found that the computational cost of PMHT is roughly the same order of magnitude as the cost of MHT and JPDA [27]. The computational cost of PMHT versus other methods has also been evaluated for multi-target tracking [28, 29]. Perlovsky's approach would likely have a computational cost similar to PMHT, since they share a similar structure, and are both based upon EM. For multi-sensor, multi-target applications, MHT and JPDA would likely require hypothesis pruning in order to avoid an exponential increase in computational cost. Since the computational cost of EM-based approaches scale only linearly with increasing numbers of sensors, it is expected that the advantages of EM-based approaches would become increasingly evident for multi-sensor applications, such as the one considered in this paper.

The paper is organized as follows. In Sections II–IV the mathematical approach is developed for the full, multi-sensor, version of the algorithm. The EM algorithm is a well-established

mathematical tool for estimating the parameters in mixture models [10–12], and could certainly be used as a framework for deriving the solution to our problem, as was done for the PMHT [13–15]. However, in this paper a different approach is offered, which Redner and Walker [12] refer to as the "traditional general approach." Here, a system of likelihood equations is derived, then solved using the partial derivatives of the optimization criterion, as described in [22], [32]. This derivation does not require an understanding of EM. In addition to the full, multi-sensor, algorithm, a simplified version is also derived in Section V which is appropriate for performing range-Doppler-only tracking from single-sensor data. In this case, the update formula for track parameters is particularly simple and efficient—it consists of a small matrix inversion for each target component. Most of the mathematical details are contained in the appendices. Appendix A contains a derivation of the parameter estimation equations for general mixture models, and also tailors these equations to the tracking model used in this paper. Appendix B contains a convergence proof which, again, is within the structure of the traditional general approach rather than depending upon EM. Finally, in Section VI results are presented from experiments designed to test the algorithm. The simplified, single-sensor version is tested on experimental radar data, while the full, multi-sensor, version is tested on synthetic data. The results, although preliminary, demonstrate robustness in the presence of nonhomogeneous clutter, and uncertainty in the number of targets present. Section VII provides a summary and discusses directions for future research.

## II. DESCRIPTION OF THE SENSORS AND DATA

The sensor model for this discussion incorporates a ground-based radar antenna having poor azimuthal resolution. In fact, we will assume the extreme case in which each sensor measures target range and Doppler (range-rate) only, but no azimuth. Thus, the track of each target can only be estimated by triangulation from multiple, spatially diverse, sensor platforms. In this work, we focus on the 2-dimensional tracking problem in which targets are assumed to lie on the zero-elevation plane, while sensors have arbitrary 3-dimensional positions.

Our method is appropriate for any collection of range/Doppler data, although as a working model it is assumed the data are acquired using a stretch receiver [33]. Here, a suite of long-duration chirps is transmitted from a stationary ground-based radar, and the corresponding suite of received signals within the coherent processing interval (CPI) is processed jointly to produce a two-dimensional digitized image in range/Doppler coordinates, referred to as a "scan frame." Examples of scan frames are shown in Figs. 2 and 3 in Section VI, where the target and clutter signatures appear as blobs of energy centered upon their true range and Doppler. The brightness (strength) of each blob is proportional to the range and RCS of the reflectors. The target and ground clutter may occupy several range-Doppler resolution cells. Of course, the range resolution is inversely proportional to the bandwidth of the transmitted signal, and is degraded by the associated pulse compression processing. Similarly, the Doppler resolution is inversely proportional to the CPI, and is degraded by sensor platform vibration, system instability, and processing. The spreading of the target signature across several range-Doppler cells may be associated with target speed and acceleration, target size, and scintillation of target cross section. Ground clutter appears in the scan frame image as a ridge centered at zero Doppler and spread across all range bins (again, refer to Figs. 2 and 3 in Section VI). The spread of the ridge beyond the Doppler resolution cell may be caused by antenna scan modulation, sensor platform vibration, as well as internal motion of the clutter (e.g., movement of leaves and branches in the wind). Depending on its width, this clutter ridge may in some cases obscure or partially obscure the signatures from targets with small radial velocities. Finally, there will be a certain amount of background noise (receiver noise) which is uniformly distributed in range/Doppler over the image.

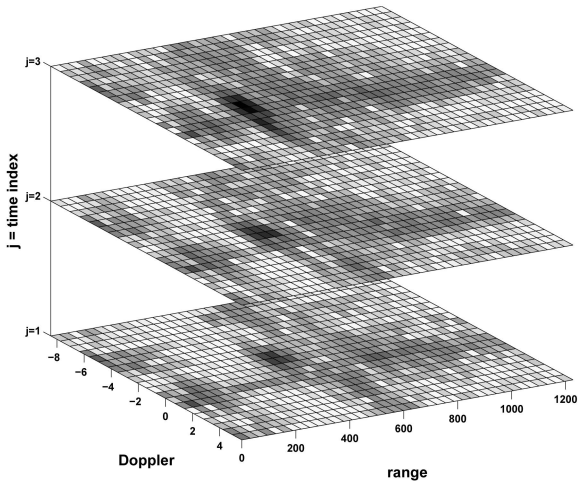


Fig. 1. Sample stack of three pixelated scan frames acquired with a particular sensor. Target signature appears as high intensity (dark) blob centered around its true range/Doppler coordinate. Since target is in motion, signature position is slightly different in each scan frame. Similar data stacks would be acquired using other sensors, and entire set of stacks from all sensors is processed jointly to estimate target tracks.

It is assumed the data are collected as follows (Fig. 1). At each time $t_j$, for $j = 1,2,3,\ldots,J$, the received signals within the CPI centered at $t_j$ are processed jointly to yield a sampled or pixelated range/Doppler image (scan frame) as described above.

Thus, for each sensor $m = 1,2,3,\ldots,M$ we acquire a stack of $J$ range/Doppler scan frames corresponding to the different times $t_j$. Note that, due to the irregular overlap in the fields-of-view of the different sensors, the signatures from some targets may not appear in the images from all sensors at all times. Each range/Doppler scan frame image is described using the notation $p_0(\mathbf{w}_{jmn})$, where $p_0$ is the pixel intensity at the range/Doppler coordinate $\mathbf{w}_{jmn} = (r_{jmn}, d_{jmn})$, which is indexed by time index $j$ (i.e., scan frame number $j$), sensor $m$, and pixel $n = 1,2,3,\ldots,N$. The total number of pixels in each scan frame is $N = N_r \times N_d$, where $N_r$ and $N_d$ are the number of bins in range and Doppler, respectively. The width of each pixel with respect to range and Doppler is $\Delta_r$ and $\Delta_d$, respectively. The data from the multiple sensor platforms are combined noncoherently to estimate target tracks, as described in the following sections.

## III. MODEL FOR THE DATA

A model for the data $p_0(\mathbf{w}_{jmn})$ can be developed which depends upon the target trajectories as well as the sensor parameters and coordinates. Suppose the coordinates of target $k$ at time $t_j$ are given by the (east, north) coordinate $(x_k(t_j), y_k(t_j)) \equiv (x_{jk}, y_{jk})$. Using a constant acceleration model, the trajectories are described by

$$x_{jk} = x_k^0 + x_k' t_j + x_k'' t_j^2 \tag{1}$$

$$y_{jk} = y_k^0 + y_k' t_j + y_k'' t_j^2. \tag{2}$$

Here $(x_k^0, y_k^0)$ is the time-zero position of target $k$, while $(x_k', y_k')$ is the time-zero velocity and $(x_k'', y_k'')$ are proportional to the $x$ and $y$ components of the acceleration. The targets are assumed to lie at zero elevation, which simplifies the discussion, as well as reducing the number of track parameters that need to be estimated. The sensors are allowed to have arbitrary 3-dimensional coordinates. Suppose there are $M$ sensor platforms, where sensor $m$ is fixed at the (east, north, elevation) coordinate $(x_m, y_m, z_m)$. Then, the range $R_{km}(t_j) \equiv R_{jkm}$ from sensor $m$ to target $k$ at time $t_j$ is given by the equation

$$R_{jkm} = \sqrt{(x_m - x_{jk})^2 + (y_m - y_{jk})^2 + z_m^2}. \tag{3}$$

The Doppler (range-rate) $D_{km}(t_j) = [\partial R_{km}(t)/\partial t]_{t=t_j}$ is then

$$\begin{aligned} D_{km}(t_j) &\equiv D_{jkm} \\ &= -\left(\frac{x_m - x_{jk}}{R_{jkm}}\right)(x_k' + 2x_k'' t_j) \\ &\quad -\left(\frac{y_m - y_{jk}}{R_{jkm}}\right)(y_k' + 2y_k'' t_j). \end{aligned} \tag{4}$$

As described in Section II, the target signatures appear as blobs of energy in the scan frame images $p_0(\mathbf{w}_{jmn})$, where recall that $\mathbf{w}_{jmn} = (r_{jmn}, d_{jmn})$ is the range and Doppler value of the $n$th pixel in the frame pertaining to the $m$th sensor at the $j$th time. The signature corresponding to target $k$ is roughly centered at the true range $R_{jkm}$ and Doppler $D_{jkm}$, measured relative to sensor $m$ at time $t_j$. We wish to construct a mathematical model that approximates these signatures, and for target $k$ we denote this model as $p_k(\mathbf{w}_{jmn} \mid \Theta_k)$, where $\Theta_k$ is the set of model parameters. In this case it makes sense to use Gaussian distributions to model the target signatures (blobs of energy), so that the signature due to target $k$ for sensor $m$ and time $t_j$ is modeled by

$$p_k(\mathbf{w}_{jmn} \mid \Theta_k) = \frac{\Delta_r \Delta_d}{2\pi\sigma_{rk}\sigma_{dk}} \exp\left\{-\frac{1}{2}\left[\left(\frac{r_{jnm} - R_{jkm}}{\sigma_{rk}}\right)^2 + \left(\frac{d_{jnm} - D_{jkm}}{\sigma_{dk}}\right)^2\right]\right\}. \tag{5}$$

Here, the set of model parameters is $\Theta_k = \{\sigma_{dk}^2, \sigma_{rk}^2, x_k^0, y_k^0, x_k', y_k', x_k'', y_k''\}$ where, in (5), the track parameters $\{x_k^0, y_k^0, x_k', y_k', x_k'', y_k''\}$ are contained implicitly within the quantities $R_{jkm}$ and $D_{jkm}$ according to (3) and (4). Also, $\Delta_r$ and $\Delta_d$ define the pixel size relative to the range and Doppler coordinates, as discussed in Section II. Note that the distribution in (5) is normalized[1] so that $\sum_{n=1}^{N} p_k(\mathbf{w}_{jmn} \mid \Theta_k) = 1$. The variance parameter $\sigma_{rk}^2$ specifies the spread (width) of the signatures with respect to the range coordinate in the scan frame image, and is related to the range resolution of the sensor and the range extent of the target. The other variance parameter $\sigma_{dk}^2$ specifies the width of the signature in Doppler, and depends both on the sensor resolution and the target motion, including internal (micro) motion. It should be noted that the model could alternatively have been formulated so that the variance parameters depend upon time index $j$ or sensor $m$, as well as sensor $k$.

The interference also requires a model, and we approximate this using two components, one for the uniform background noise (receiver noise), and one for the stationary clutter. We designate the two interference indices as $k = (K-1)$, $K$ and the target indices as $k = 1,2,3,\ldots,(K-2)$. The background noise (receiver noise) component $k = K-1$, uniform (constant) in both range and Doppler, is described by

$$p_{K-1}(\mathbf{w}_{jmn} \mid \Theta_{K-1}) = \frac{1}{N} \tag{6}$$

[1] Actually, the normalization is approximate, and assumes the pixel dimensions $(\Delta_r, \Delta_d)$ are small relative to $(\sigma_{rk}, \sigma_{dk})$.

where $N$ is the number of pixels in each range/Doppler image, as discussed above. The $1/N$ factor is necessary for normalization, i.e., to insure $\sum_{n=1}^{N} p_k(\mathbf{w}_{jmn} \mid \Theta_k) = 1$. Note, here the set of parameters $\Theta_{K-1}$ is empty. The other interference component $k = K$, which models the reflected energy from stationary ground clutter, is uniform (constant) in range, and zero-mean Gaussian in Doppler, i.e.,

$$p_K(\mathbf{w}_{jmn} \mid \Theta_K) = \frac{\Delta_d}{\sqrt{2\pi}\sigma_{dk}N_r} \exp\left\{-\frac{1}{2}\left(\frac{d_{jnm}}{\sigma_{dK}}\right)^2\right\}. \tag{7}$$

Here, $N_r$ is the number of range bins in each pixel array, as discussed in the previous section. Note, here the set of parameters is $\Theta_K = \{\sigma_{dK}^2\}$. A further discussion of clutter modeling is given in [22].

The total model for the image data is the weighted summation (mixture) of individual target and clutter components

$$p(\mathbf{w}_{jmn} \mid \Theta) = \sum_{k=1}^{K} E_{km} p_k(\mathbf{w}_{jmn} \mid \Theta_k). \tag{8}$$

Here, $E_{km}$ is the relative weighting for each of the model components, which we refer to as the mixture weight. For target components, the mixture weight is proportional to the target's RCS and range as viewed by sensor $m$. Since RCS can be strongly dependent upon aspect angle, we allow $E_{km}$ to depend upon both target $k$ and sensor $m$. This dependence upon sensor $m$ also allows for the fact that the fields-of-view for the different sensors overlap in an irregular manner, so that a signature from a certain target may not appear in the data from all sensors. In (8), the total set of model parameters is denoted by $\Theta = \{E_{km}, \Theta_k\}$, where $k = 1,2,\ldots,K$ and $m = 1,2,\ldots,M$.

The well-known problem of estimating the best number of components $K$ in mixture models is discussed in the literature. For example, mathematical strategies are described in [34], [35], [36] and the references therein. In our application, this so-called "model selection" problem stems from the fact that the true number of targets is not known in most contexts, and therefore we must make an educated guess for the appropriate number of model components $K$. If there are more model components than targets, unneeded components can presumably be eliminated automatically as their mixture weights $E_{km}$ adapt to small values. A tracking example is presented in Section VI where this type of process is illustrated. In cases where the true number of targets exceeds the number of model components, the behavior of the model during its adaptation is not as well understood. It is speculated that either the process will lock onto $K-2$ targets, ignoring the rest, or some model components will lock onto multiple targets sharing similar parameters. In a future study, we will investigate how model selection strategies described in the literature can be adapted to the tracking problem, for choosing the number of target components, pruning unneeded components, and adding components when necessary.

A useful result can be obtained by summing both sides of (8) over the pixel index $n$, and using the (previously stated) fact that $\sum_{n=1}^{N} p_k(\mathbf{w}_{jmn} \mid \Theta_k) = 1$, to obtain

$$\sum_n p(\mathbf{w}_{jmn} \mid \Theta) = \sum_k E_{km} \tag{9}$$

for each $j = 1,2,\ldots,J$. Here we have assumed that if a target is in the field-of-view of a particular sensor $m$, it will remain in the field-of-view for the entire set of scan frames $j = 1,2,\ldots,J$. Thus, the sum of the relative model weights is independent of $j$. It is sensible to place a constraint so that, for each scan frame $j$ and sensor $m$, the total energy in the model equals the total energy in the measured data, i.e., $\sum_n p(\mathbf{w}_{jmn} \mid \Theta) = \sum_n p_0(\mathbf{w}_{jmn})$. Combining this constraint with (9), we obtain the following constraint on the mixture weight parameters, valid for every scan frame $j$ and sensor $m$:

$$\sum_n p(\mathbf{w}_{jmn} \mid \Theta) = \sum_n p_0(\mathbf{w}_{jmn}) = \sum_k E_{km}. \tag{10}$$

## IV. PARAMETER ESTIMATION

The goal is to find the set of model parameters $\Theta = \{E_{km}, \Theta_k\}$ providing the best match between the the model $p(\mathbf{w}_{jmn} \mid \Theta)$ and the data $p_0(\mathbf{w}_{jmn})$. The model, described in Section III, is completely specified by the the mixture weights $E_{km}$, the variances $\sigma_{dk}^2$ and $\sigma_{rk}^2$, and the parameters $\{x_k^0, y_k^0, x_k', y_k', x_k'', y_k''\}$ describing the target trajectories. In this paper the Einsteinian log-likelihood [22, ch. 4.4.1]

$$\begin{aligned} LL(\Theta) &= \sum_{j,m,n} p_0(\mathbf{w}_{jmn}) \ln p(\mathbf{w}_{jmn} \mid \Theta) \\ &= \sum_{j,m,n} p_0(\mathbf{w}_{jmn}) \ln \sum_{k=1}^{K} E_{km} p_k(\mathbf{w}_{jmn} \mid \Theta_k) \end{aligned} \tag{11}$$

will serve as a criterion to quantify the similarity between the model and the data. Given the constraint in (10), it can be shown that $LL$ will be maximized when the model and the data match, i.e., for $p(\mathbf{w}_{jmn} \mid \Theta) = p_0(\mathbf{w}_{jmn})$. Note that maximizing $LL$ is equivalent to minimizing the cross-entropy, a.k.a, the Kullback-Leibler (KL) divergence, a well-established metric from the information theory arena [41]. Mathematical and physical justifications for both Einsteinian log-likelihood and the KL divergence have been discussed in detail [22, 37–40]. It should be emphasized that the tracking algorithm described

here is a "batch" processing approach, i.e., the frames from all time indices $j$ are processed jointly. This is apparent in (11) since the log-likelihood includes a summation over $j$.

The mathematical details of the optimization are contained in Appendix A. Briefly, a system of equations is derived which is satisfied by the parameters maximizing $LL$, subject to the constraint of (10). Unfortunately, an analytical solution is intractable since the system of equations is large, coupled, and nonlinear. However, iterative techniques can be employed to solve the system of equations and, in Appendix A, an efficient recursive technique is described which is a special case of EM. This technique defines a recursive update formula for each parameter. For example, using the notation $E_{km}^{(I)}$ and $\Theta_k^{(I)}$ to indicate the estimates of the parameters at the $I$th iteration, the recursive update formulas for $E_{km}$, $\sigma_{dk}^2$, and $\sigma_{rk}^2$ are

$$E_{km}^{(I+1)} = \frac{1}{J}\sum_{j,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \tag{12}$$

$$(\sigma_{dk}^2)^{(I+1)} = \frac{\sum_{j,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn)(d_{jnm} - D_{jkm})^2}{\sum_{j,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn)} \tag{13}$$

and

$$(\sigma_{rk}^2)^{(I+1)} = \frac{\sum_{j,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn)(r_{jnm} - R_{jkm})^2}{\sum_{j,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn)} \tag{14}$$

where

$$\begin{aligned} P^{(I)}(k \mid jmn) &= \frac{E_{km}^{(I)} p_k(\mathbf{w}_{jmn} \mid \Theta_k^{(I)})}{\sum_{k'} E_{k'm}^{(I)} p_{k'}(\mathbf{w}_{jmn} \mid \Theta_{k'}^{(I)})} \\ &= \frac{E_{km}^{(I)} p_k(\mathbf{w}_{jmn} \mid \Theta_k^{(I)})}{p(\mathbf{w}_{jmn} \mid \Theta^{(I)})}. \end{aligned} \tag{15}$$

There is an analogous rule for updating the tracking parameters $\{x_k^0, y_k^0, x_k', y_k', x_k'', y_k''\}$, which is described later in this section. The equations above imply starting with an initial guess for the parameters $\{E_{km}^{(0)}, \Theta_k^{(0)}\}$, then alternating between updating $P^{(I)}(k \mid jmn)$ using (15), then updating the parameters $\{E_{km}^{(I+1)}, \Theta_k^{(I+1)}\}$ using (12)–(14). In fact, these steps correspond to the E-step and the M-step, respectively, of the EM algorithm. This iterative procedure is guaranteed to converge to a local maximum of $LL$, as shown in Appendix B. Note that the mixture weights $E_{km}$ are updated using (12) for all target and clutter components $k = 1, 2, \ldots, K$. In fact, (12) is appropriate for updating the weights of arbitrary mixture models, not only the tracking model considered here. The variances $\sigma_{dk}^2$ and $\sigma_{rk}^2$ are updated for the appropriate model components, as indicated by the target and clutter model equations. Since the variance parameters are initialized to large values, the components are initially fuzzy and indistinct. However, with increasing iterations, the variances tend to adaptively decrease, and the individual components gradually converge and lock on to the target signatures, or to portions of the clutter. Thus, the model iteratively adapts to fit the data, as we demonstrate with the results in Section VI.

If the model is viewed probabilistically [22, ch. 4.4.8], then (15) is simply a form of Bayes' rule, and the quantities $P^{(I)}(k \mid jmn)$ can be construed as the probabilities that the energy in pixel $(j,m,n)$ originates from target or clutter component $k$. Therefore, $P^{(I)}(k \mid jmn)$ are referred to as the "association probabilities" [22]. Equation (12) therefore makes intuitive sense—it simply states that $E_{km}^{(I+1)}$ is the sum of all pixel values $p_0(\mathbf{w}_{jmn})$, where the pixels are weighted by their association probabilities. Similarly, (13) and (14) correspond to the usual definition of sample variance, with the distinction that the pixels are weighted by their association probabilities. From (15) it is apparent that

$$\sum_k P^{(I)}(k \mid jmn) = 1 \tag{16}$$

for any iteration $I$.

For compactness, we define the angled bracket notation

$$\langle * \rangle^{(I)} \equiv \sum_{j,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn)(*) \tag{17}$$

where the asterisk $*$ denotes a generic quantity. Thus, (12)–(14) can be rewritten in the compact form

$$E_{km}^{(I+1)} = \frac{\langle 1 \rangle^{(I)}}{J} \tag{18}$$

$$(\sigma_{dk}^2)^{(I+1)} = \frac{\sum_m \langle (d_{jnm} - D_{jkm})^2 \rangle^{(I)}}{\sum_m \langle 1 \rangle^{(I)}} \tag{19}$$

and

$$(\sigma_{rk}^2)^{(I+1)} = \frac{\sum_m \langle (r_{jnm} - R_{jkm})^2 \rangle^{(I)}}{\sum_m \langle 1 \rangle^{(I)}}. \tag{20}$$

The converged values of Doppler variance parameters $\sigma_{dk}^2$ are related to the radar Doppler resolution and target dynamics, and are thus difficult to estimate a priori. On the other hand, the converged values of the range variance parameters $\sigma_{rk}^2$ are related to the radar range resolution, modified by the range extent of the target, and are assumed to be better known a priori. Therefore, rather than adaptively estimating range variance using (20), it may be advantageous to evolve $\sigma_{rk}$ according to a predetermined schedule [22, 25, 26]. For example, $\sigma_{rk}$ can be initialized to a rather large value, so that the target components can "see" large sections of the pixel data, then decreased according to an exponential decay during EM iterations to a steady state value corresponding to the appropriate sensor resolution.

Finally, we derive the recursive update formula for each of the tracking parameters $\{x_k^0, y_k^0, x_k', y_k', x_k'', y_k''\}$, $k = 1,2,3,\ldots,(K-2)$. In the single-sensor case, described below in Section V, the update formula for these parameters is a simple closed-form expression, analogous to (18)–(20). Unfortunately, a closed-form update formula cannot be derived for the general multi-sensor case. However, local convergence is still guaranteed (see Appendix B) if the tracking parameters are simply "nudged" along the uphill gradient of $LL$ during each iteration. This actually corresponds to a special case of the generalized EM (GEM) procedure [11]. Using $s_k$ as a generic placeholder for any one of the tracking parameters $\{x_k^0, y_k^0, x_k', y_k', x_k'', y_k''\}$, the update formula is given by

$$s_k^{(I+1)} = s_k^{(I)} + h \cdot \left[\frac{\partial LL}{\partial s_k}\right] \tag{21}$$

where $h$ is the gradient ascent stepsize, and

$$\frac{\partial LL}{\partial s_k} = \sum_m \left\langle \left(\frac{r_{jnm} - R_{jkm}}{\sigma_{rk}^2}\right)\left(\frac{\partial R_{jkm}}{\partial s_k}\right)\right\rangle^{(I)} + \sum_m \left\langle \left(\frac{d_{jnm} - D_{jkm}}{\sigma_{dk}^2}\right)\left(\frac{\partial D_{jkm}}{\partial s_k}\right)\right\rangle^{(I)}. \tag{22}$$

This expression is derived in Appendix A. It is straightforward to compute explicit expressions for $\partial R_{jkm}/\partial s_k$ and $\partial D_{jkm}/\partial s_k$ for each of the parameters $s_k = \{x_k^0, y_k^0, x_k', y_k', x_k'', y_k''\}$ using (3) and (4). For example, $\partial R_{jkm}/\partial x_k^0 = -(x_m - x_{jk})/R_{jkm}$. Although (21) describes a single gradient ascent step during each iteration, multiple steps can be taken during each iteration while holding the association probabilities $P(k \mid jmn)$ fixed. In practice, the convergence rate can be optimized, using trial-and-error to determine a suitable combination of stepsize $h$ and steps/iteration.

Although the present study is mainly concerned with tracking, it is important to consider the issue of automatic detection and target declaration. The standard detection approach, e.g., described in [22, sect. 7.2.9], utilizes a log-likelihood ratio test which operates on the converged parameter values. The issue of detection will be studied in more detail in the future.

## V. RANGE-ONLY TRACKING FROM A SINGLE SENSOR

If only a single sensor platform is available, and measurements of azimuth are unavailable, it may still be desirable to perform range-Doppler-only tracking. Here, the target and interference (clutter plus noise) models given by (5)–(7) remain valid, as well as the total model for the data given by (8). Also, the variance $(\sigma_{dk}^2, \sigma_{rk}^2)$ and mixture weight $E_{km}$ parameters are still updated using (18)–(20). However, in the single-sensor case the target track parameters can be updated using a closed-form formula, which is simpler and more efficient than the gradient ascent procedure described by (21) used for the multi-sensor case. Note that the analysis in this section is very similar to the analysis given in [22, ch. 7.2.6]. However, whereas the the previous work utilized a constant velocity model, here a constant acceleration model is used which is slightly more complicated.

Suppose the range between the targets and the (single) sensor can be described using a constant acceleration model, i.e.,

$$R_{jkm} = R_k^0 + R_k' t_j + R_k'' t_j^2 \tag{23}$$

where $R_k^0$ is the time-zero range of target $k$, $R_k'$ is the time-zero range-rate, and $R_k''$ is proportional to the range-acceleration. Although there is only a single sensor $m = 1$, the $m$ subscript is maintained in the quantity $R_{jkm}$ for consistency with the expression for the target model in (5). The Doppler (range-rate) at time $t_j$ is the derivative of range with respect to time evaluated at $t_j$, i.e.,

$$D_{jkm} = R_k' + 2R_k'' t_j. \tag{24}$$

In Appendix A, an update formula is derived for iteratively adjusting the values of $\{R_k^0, R_k', R_k''\}$ in order to maximize $LL$ which, as in the multi-sensor case, is described by (11). This update formula is described by the following $3 \times 3$ set of linear equations:

$$\bar{\mathbf{H}}_k \begin{Bmatrix} (R_k^0)^{(I+1)} \\ (R_k')^{(I+1)} \\ (R_k'')^{(I+1)} \end{Bmatrix} = \begin{Bmatrix} \langle r_{jnm}\rangle/\sigma_{rk}^2 \\ \langle r_{jnm} t_j\rangle/\sigma_{rk}^2 + \langle d_{jnm}\rangle/\sigma_{dk}^2 \\ \langle r_{jnm} t_j^2\rangle/\sigma_{rk}^2 + \langle 2d_{jnm} t_j\rangle/\sigma_{dk}^2 \end{Bmatrix} \tag{25}$$

where

$$\bar{\mathbf{H}}_k = \frac{1}{\sigma_{rk}^2}\begin{Bmatrix} \langle 1\rangle & \langle t_j\rangle & \langle t_j^2\rangle \\ \langle t_j\rangle & \langle t_j^2\rangle & \langle t_j^3\rangle \\ \langle t_j^2\rangle & \langle t_j^3\rangle & \langle t_j^4\rangle \end{Bmatrix} + \frac{1}{\sigma_{dk}^2}\begin{Bmatrix} 0 & 0 & 0 \\ 0 & \langle 1\rangle & \langle 2t_j\rangle \\ 0 & \langle 2t_j\rangle & \langle 4t_j^2\rangle \end{Bmatrix}.$$

Here, for example, $(R_k^0)^{(I+1)}$ denotes the updated estimate for $R_k^0$ computed at the $(I+1)$th iteration. For the sake of compactness, the $(I)$ superscript has been omitted from the angled bracket notation $\langle * \rangle$ used for the elements in the above matrices. However it should be understood that here

$$\langle * \rangle \equiv \sum_{j,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn)(*). \tag{26}$$

It can be shown that, in the absence of Doppler measurements, the set of equations (25) is equivalent to second-order polynomial regression in which data samples are weighted by their association probabilities.

Unlike the update formulas for $E_{km}$, $\sigma_{dk}$, and $\sigma_{rk}$, given by (18)–(20), the matrix expression

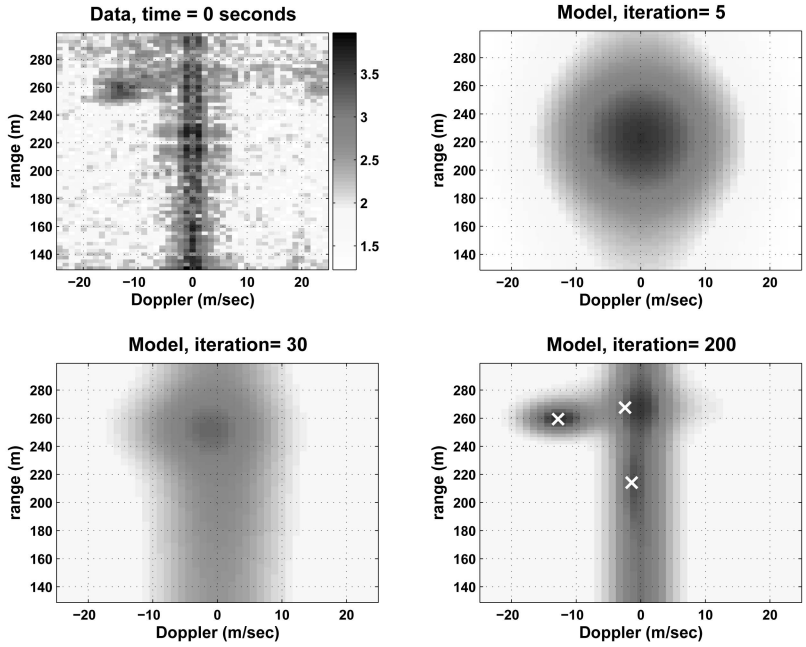


Fig. 2. Upper left: scan frame data acquired at time $t = 0$ s in which a single target signature appears at Doppler and range values of roughly ($-12$ m/s) and (258 m), respectively. Remaining three plots show evolution of model, which adapts to fit data as iterations increase. White Xs in lower right plot indicate converged range/Doppler coordinates $R_k^0$ and $R_k'$ for target components in model.

(25) actually represents a set of three coupled update formulas for the three tracking parameters $\{R_k^0, R_k', R_k''\}$. Upon each iteration, these parameters are updated by inverting (25). Note that a separate $3 \times 3$ matrix inversion is performed for each target component $k = 1, 2, \ldots, (K-2)$.

## VI. RESULTS

In this section, we describe results when the algorithm is applied to two cases. First, the algorithm is tested against experimental stretch radar data [33] from a single sensor using only the range and Doppler data from the sensor. The single sensor results are presented as a proof of concept and to illustrate the performance of the algorithm in realistic, inhomogeneous clutter, variable signal to interference, and accelerating and maneuvering targets. In the second case, we consider multiple sensors viewing the same scene. The sensors are assumed to have poor or no azimuth resolution but good range and Doppler resolution. Combining the data from multiple sensors offers the opportunity to more accurately locate and track targets using triangulation of the sensor data if the difficult problem of target association can be solved. In the second case we present results from computational experiments designed to test the algorithm since experimental multi-sensor data are not available to us.

Let us now describe the single-sensor results computed from experimental data, using the simplified version of the algorithm described in Section V. These results will be mainly qualitative since neither true target tracks nor detailed information about the radar were provided. Nevertheless, this analysis serves as a useful proof-of-concept for the algorithm. The experimental setup consisted of a stretch receiver mounted on a tower overlooking a wooded area, and data were collected as various targets moved through the area. The raw data were converted to a suite of range/Doppler scan frames using standard processing techniques [33], as discussed in Section II. The algorithm described in Section V was used to jointly process multiple sets of scan frames, where each scan frame is computed from data centered around a different time $t_j$, $j = 1, 2, \ldots, J$.

The upper left plot of Fig. 2 shows a single scan frame acquired at a reference time of $t = 0$ s. A target signature appears as a "blob of energy" at Doppler and range values of roughly ($-12$ m/s) and (258 m), respectively. There is a significant ridge of ground clutter from the stationary background centered at zero Doppler, extending across all range bins. There is also a noise background which is roughly uniform in range/Doppler, although with some significant inhomogeneities. For the model we used five components: three target components, one uniform background noise component, plus one clutter component shaped like a ridge centered at zero Doppler.

The remaining three plots of Fig. 2 show how the model evolves with increasing iterations. Since the model variances are initialized to large values, the model is initially broad, fuzzy, and indistinct (upper right). However, as the iterations progress, the parameters adapt in such a way as to eventually define

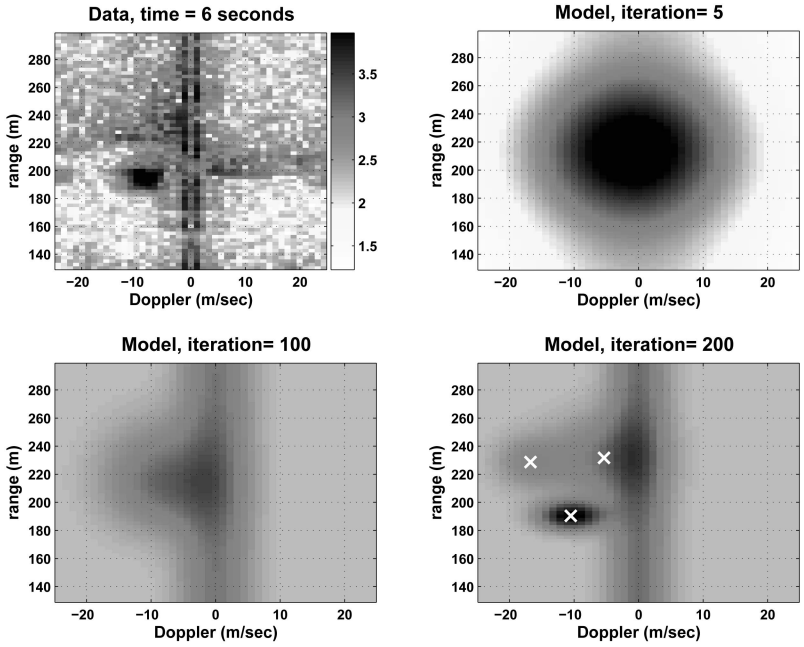


Fig. 3. Same as Fig. 2, however here scan frame was acquired at a later time of $t = 6$ s. Target signature has now moved to Doppler and range values of roughly (−9 m/s) and (195 m), respectively.

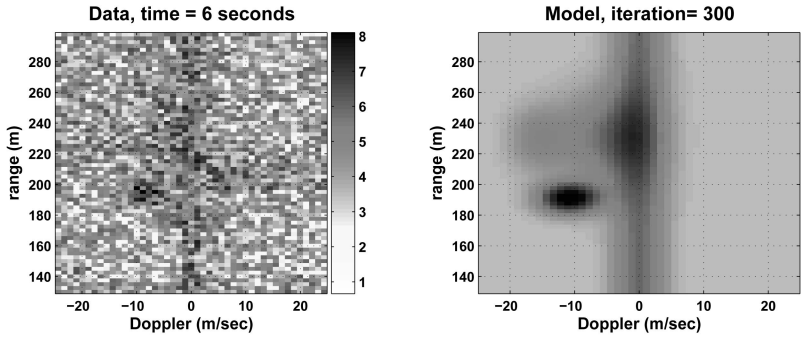


Fig. 4. Left-hand plot shows same scan frame shown in Fig. 3, however a large amount of artificial noise was added which almost completely obscures target signature. Nevertheless (right-hand plot), one of the model components was able to adapt to range/Doppler coordinate in close vicinity of target signature.

a model which shares most of the significant features of the data. The lower right plot in Fig. 2 shows the converged model after 200 iterations. The white Xs indicate the converged range/Doppler estimates $R_k^0$ and $R_k'$ for the three target components. Notice that one of the three target components has properly converged to the true target signature, while the extra two target components have converged to portions of the inhomogeneous background clutter. Although Fig. 2 shows the data and model corresponding to only a single scan frame, it is important to note that the results were computed by jointly processing 10 scan frames, starting at $t = 0$ and proceeding in intervals of 0.25 s.

The upper left plot of Fig. 3 shows scan frame data corresponding to a different time, $t = 6$ s. The target signature has now moved to Doppler and range values of roughly (−9 m/s) and (195 m), respectively. Here, the inhomogeneities in the background are even more extreme than for the $t = 0$ case shown in Fig. 2. However, the model was able to adapt quite nicely, with one of the target components locking on to the true target signature, and the extra components adapting to fit the clutter.

In Figs. 2 and 3, the target signatures were rather strong, with signal-to-interference ratios of roughly 3 dB (computed from the peak value of the signature relative to an average background value in the image frame). Next, the tracking experiment was repeated for the same case shown in Fig. 3. However the problem was intentionally made more difficult by adding a large amount of artificial noise, so that the signal-to-interference ratio was reduced to roughly 0 dB. This noise is uniformly distributed in range and Doppler, and is signal-independent, i.e., it was simply added to the intensity image. The data and the converged model for this noisy experiment are shown in Fig. 4, which shows that one of the model

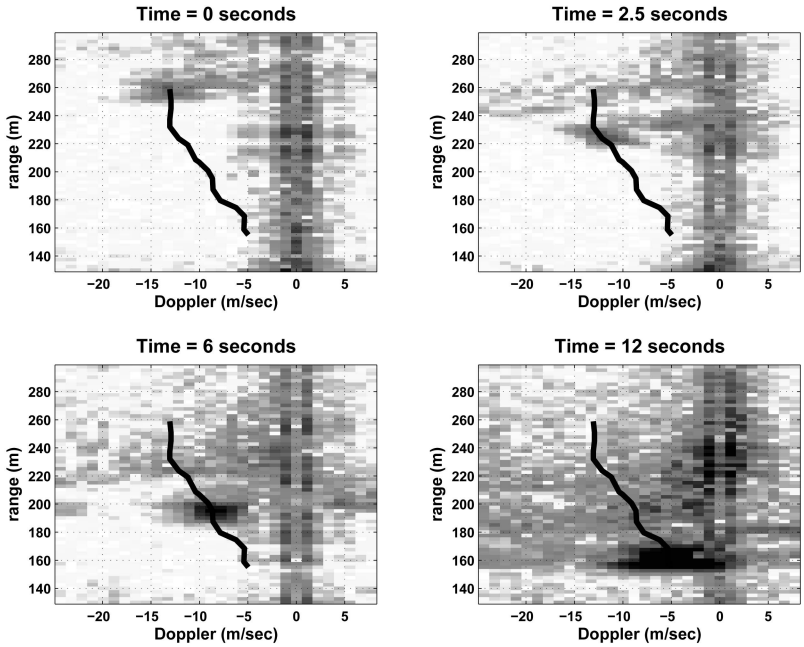


Fig. 5. Tracking over extended time interval for smoothly decelerating target. Each scan frame corresponds to different time instance with 12 s interval. Overlaid is dark line showing track computed using our method.

components locked on to the target signature in the data, despite the fact that the target signature is almost completely obscured by noise.

Although the present study is mainly concerned with tracking, it is important to consider the issue of automatic detection and target declaration. For example, in the converged model shown in the lower right plot of Fig. 3 there are three target components which converged to the three different range/Doppler coordinates indicated with white Xs. One component has locked on to the target signature, while the other two have adapted to fit portions of the clutter. Here, the detection problem consists of automatically declaring the true target, while disregarding the other two components.

The standard detection approach, e.g., described in reference [22, sect. 7.2.9], utilizes a log-likelihood ratio test evaluated at the converged track and variance parameter values. In the examples shown thus far, it appears that perhaps a simple detection rule could be devised based upon the compactness of the target component, i.e., based upon the converged values of the variance parameters $\sigma_{dk}$ and $\sigma_{rk}$. The issue of detection will be studied in more detail in a future publication.

Next, sample results are presented in which tracking was performed over longer time intervals. Here, a sliding window approach was used to estimate tracks along extended, irregular, paths in the range/Doppler space. Overlapping sets ("batches") of 6 scan frames were used to compute each point along the path. For example, the first point on the path was computed by jointly processing 6 scan frames within the time interval $0 \leq t_j \leq 1.25$ s, the second point on the path was computed by jointly processing 6 scan frames within the time interval $0.5 \leq t_j \leq 1.75$ s, etc. Fig. 5 shows results from tracking a smoothly decelerating target. Four different scan frames are shown corresponding to times of $t = 0$, 2.5, 6, and 12 s. Superimposed on the images is a dark line which indicates the extended track computed using the sliding window method over a 12 s interval. Notice the close agreement between the location of the target signatures and the computed track. Next, Fig. 6 shows results from tracking a maneuvering target. Here the target accelerates significantly over the time observation interval, whereas previously the target velocity changed relatively slowly. Each scan frame in Fig. 6 corresponds to a different time instance within a 15.5 s interval. Overlaid is a dark line showing the track computed using the sliding window method which, again, shows close agreement between the location of the target signatures and the computed track.

In the remainder of this section results are presented for the full, multi-sensor, version of the algorithm described in Section IV. Since experimental multi-monostatic data were unavailable to us, the results are based on sets of synthetic range/Doppler scan frames of the type that would be produced from UHF stretch radar data [33]. In these simulations, emphasis was more on data association than on detection performance. Therefore, the data were generated with a relatively high signal-to-interference ratio of approximately 8 dB, computed from the peak value of the signature relative to an average background value in the image frame. However, the zero-Doppler ridge from stationary clutter was

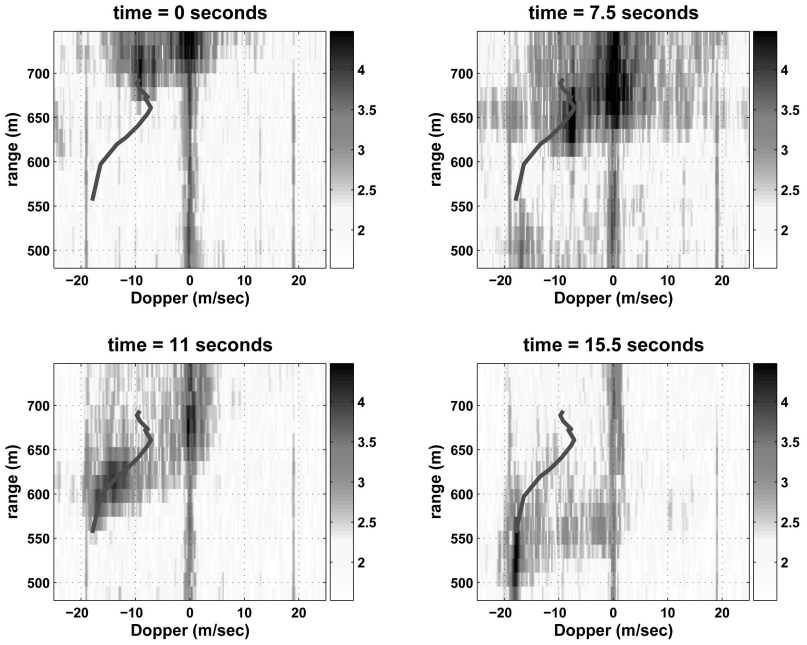


Fig. 6. Tracking over extended time interval for maneuvering target. Each scan frame corresponds to different time instance within 15.5 s interval. Overlaid is dark line showing track computed using our method.

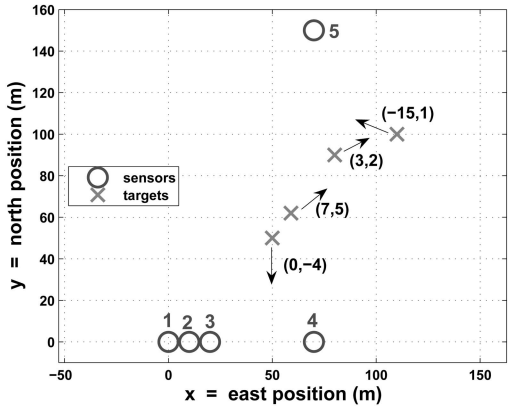


Fig. 7. Geometry for multi-sensor simulation. Each arrow shows direction of motion for corresponding target, and numbers in parentheses are $x$ (east) and $y$ (north) components of velocity $(x'_k, y'_k)$.

generated with roughly the same peak amplitude as the peak amplitude of the target signatures. Therefore this ridge overlaps and obscures the target signatures having low Doppler, as illustrated in Fig. 8. The synthetic data were generated by placing "blobs" of energy at appropriate target and clutter locations at each frame time without errors or biases, other that those associated with the dimensions of the "blobs" which are related to system resolution and target spreading effects. While errors and biases may be important in the ultimate application, we want to illustrate by this example the native data association performance of the proposed multi-sensor algorithm with multiple targets and sensors.

Fig. 7 shows the geometry for the experiment. There are 5 sensors and 4 targets, two of which have constant velocity and two of which are turning and accelerating. These four targets have $x$ (east) and $y$ (north) components of their velocities given by $(x'_k, y'_k) = (0,-4),(7,5),(3,2),(-15,1)$ m/s, and corresponding acceleration components proportional to $(x''_k, y''_k) = (-0.4,1),(-2,0),(0,0),(0,0)$ m/s$^2$. Sensors 1–3 are spaced roughly 10 meters apart,[2] while the positions of sensors 4 and 5 are more widely spaced. The spatial diversity is exploited to "triangulate" accurate track estimates for the targets. The targets were observed by all five sensors at 4 different times separated by 0.25 s. During the 4 frames of data the targets move a distance comparable to the sensor range resolution. Thus, the algorithm was required to jointly process a total of twenty scan frames in order to estimate target tracks. Fig. 8 shows four of these twenty scan frames, corresponding to sensors 1 and 5 at times $t_1 = 0$ and $t_4 = 0.75$ s. From this figure it is apparent that the data association problem is not trivial. For example, it is not obvious "by eye" which signatures in the bottom-right plot of Fig. 8 correspond to which signatures in the top-left plot.

The model was constructed of 8 components, i.e., 6 target components plus a uniform noise component, plus a zero-Doppler clutter ridge component. We assumed no prior knowledge about how many targets were actually present, and therefore 2 more target components were used than actual targets present. The code seemed to converge most quickly by starting off with 30 iterations using data only from sensors 1–3 (these sensors are relatively close together),

[2]Sensors 1–3 do not operate as an array since their data are processed noncoherently.

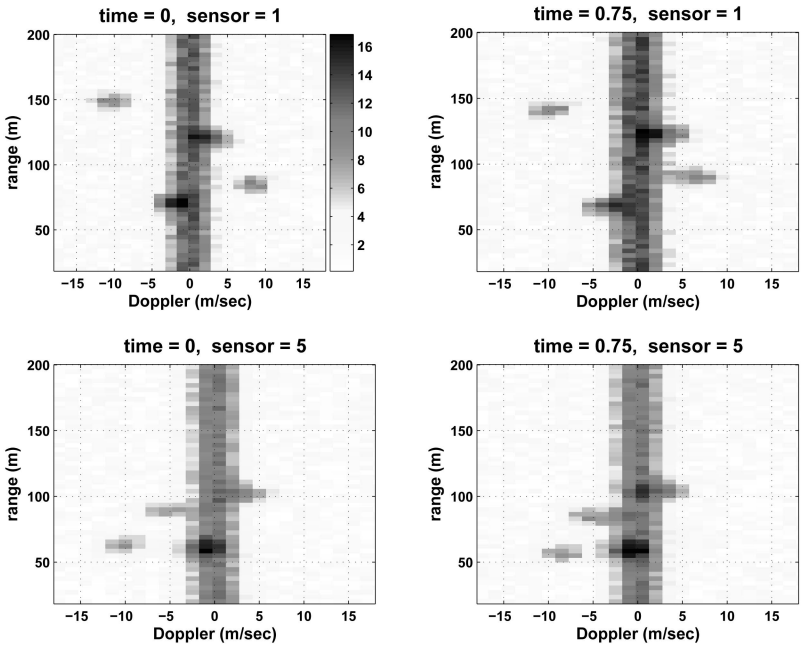


Fig. 8. Synthetic range/Doppler image data showing scan frames for sensor 1 (time = 0, 0.75 s) and sensor 5 (time = 0, 0.75 s). There are 4 target signatures present. Note that signatures of lower velocity targets significantly overlap clutter ridge centered at zero Doppler.

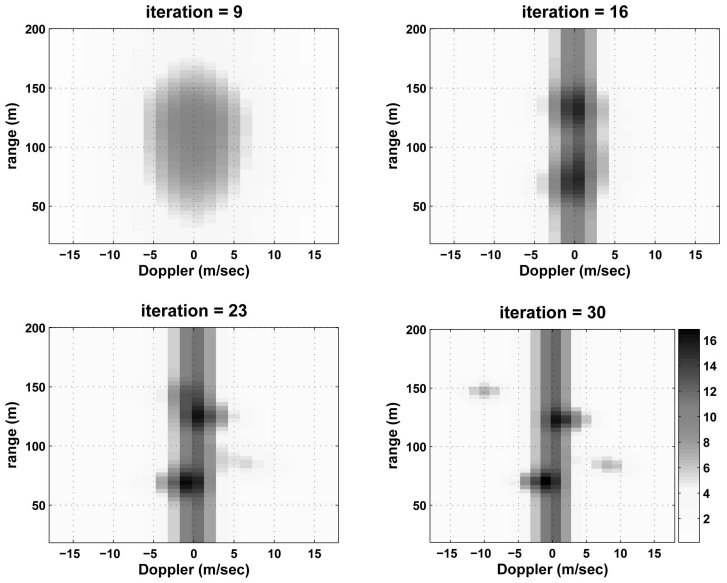


Fig. 9. Model evolution for sensor 1, time = 0 data. Compare converged model (bottom-right plot here) to actual data (top-left plot of Fig. 8).

followed by 80 additional iterations after adding the data from sensors 4 and 5. Note that for each of these iterations, the tracking parameters were adjusted using 50 gradient ascent steps, as discussed in Section V.

The evolution of the model is shown in Fig. 9 for the scan frame acquired by sensor 1 at $t_1 = 0$ s. The model is initially fuzzy and indistinct (top-left plot), but after 30 or more iterations it converges to an accurate approximation of the data, as can be seen by comparing the bottom-right plot in Fig. 9 with the top-left plot in Fig. 8. In Fig. 10 it is shown how the parameter values evolve during the adaptation of the model. Separate plots are given for the east ($x$) components of time-zero velocity $x'_k$, time-zero position $x^0_k$, and acceleration $x''_k$. Similar plots were obtained for the $y$ (north) components $y'_k$, $y^0_k$, and $y''_k$, but they are not shown here. In Fig. 10, a plot is also given showing the evolution of the mixture weights $E_k$ for each of the target components ($E_k$ of clutter components are not shown).

As mentioned above, the first 30 iterations were performed using data only from sensors 1–3 while the remaining iterations were performed using all data from sensors 1–5. There are curves in the

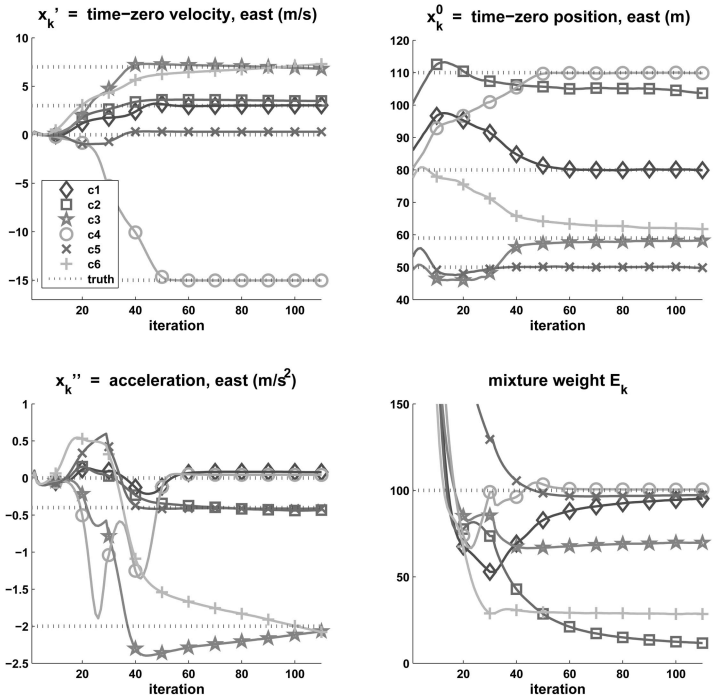


Fig. 10. Evolution of parameter estimates with iterations. Notations c1, c2, etc., in legend refer to target components 1–6 used in model. Although we only show plots for $x$ (east) components of position, velocity, and acceleration, similar plots were obtained for $y$ (north) components.

plots of Fig. 10 for each of the 6 target components labelled "c1–c6," and these are compared with the true parameter values (dashed lines). Recall, there are only 4 targets actually present. From the plots, it can be seen that components c1, c4, and c5, converge asymptotically to the true target parameters. In contrast, component c2 seems to drift around without locking onto a target, and its corresponding mixture weight parameter decreases toward zero with increasing iterations. Based upon its dwindling $E_k$ value, component c2 could easily be pruned from the model. The most interesting case involves components c3 and c6, which seem to lock onto the same target, as shown by the fact that their velocity, position, and acceleration estimates converge together in order to share a true target track. The mixture weight parameter $E_k$ plot (right-most plot) shows that the weights for c3 and c6 roughly sum up to the true mixture weight of 100. Presumably, it would be straightforward to detect this condition in which two components lock onto the same target. Then one of the two could simply be pruned from the model if necessary. It is an open question how the model would adapt for cases in which the true number of targets exceeds the number of model components. Strategies for handling this so-called "model selection" problem for general mixture models are described in [34], [35], [36] and the references therein.

## VII. CONCLUSIONS

In this paper we describe a mathematical algorithm for robust multi-target tracking from multiple radar platforms, for difficult cases in which measurements of azimuth are unavailable. A simplified version of the algorithm is also presented which is appropriate for tracking from a single radar platform.

The single-sensor version of the algorithm is tested on experimental data. These results show the approach to be very promising, and robust in the presence of significant, inhomogeneous, background clutter. The full, multi-sensor, algorithm is tested on synthetic data. These results demonstrate that accurate tracks can be estimated by exploiting spatial diversity in the sensor locations. Furthermore, the algorithm appears to be robust in the presence of significant clutter, and uncertain knowledge regarding the number of targets present.

It should be emphasized that while the initial results are promising, they are only preliminary. Significant issues require further study in order to make the algorithm practical. For example, it is well known that EM can converge to local maxima in the optimization criterion, and therefore initialization will likely affect the converged solution. We still need to determine how often these "traps" will occur, under which conditions they can cause the algorithm to fail, and how we can initialize the algorithm to prevent them. Also, since the true number of targets is seldom known a priori, more work is needed to develop strategies for choosing the number of target components, pruning unneeded components, and adding components when necessary. Furthermore, automatic detection strategies and performance will need to be studied to judge our ability to distinguish

targets from nonhomogeneous clutter. Finally, more study is required to determine the robustness of the algorithm in the presence of sensor platform errors and vibrations.

## APPENDIX A. DERIVATION OF PARAMETER ESTIMATION EQUATIONS

Here the equations are derived which are satisfied by the parameters maximizing $LL$ subject to a constraint. The general procedure is first developed, valid for an arbitrary mixture model. These results are then used to derive the particular equations corresponding to the tracking model. Good references for the general procedure described here are Perlovsky's book [22], and the seminal work by Duda and Hart [32]. The notation we use is very similar to [32].

First, consider maximizing $LL$ using a general mixture model. One might seek to maximize $LL$ directly, by finding values for the set of parameters $\Theta_k$ for which all partial derivatives $\partial LL/\partial\Theta_k$ are zero. Using the expression for $LL$ from (11),

$$\begin{aligned}\frac{\partial LL}{\partial\Theta_k} &= \sum_{j,m,n} p_0(\mathbf{w}_{jmn})\frac{\partial}{\partial\Theta_k}\left[\ln\sum_{k'} E_{k'm}p_{k'}(\mathbf{w}_{jmn}\mid\Theta_{k'})\right]\\ &= \sum_{j,m,n}\left[\frac{p_0(\mathbf{w}_{jmn})}{\sum_{k'}E_{k'm}p_{k'}(\mathbf{w}_{jmn}\mid\Theta_{k'})}\right]\frac{\partial}{\partial\Theta_k}[E_{km}p_k(\mathbf{w}_{jmn}\mid\Theta_k)]\\ &= \sum_{j,m,n} p_0(\mathbf{w}_{jmn})\left[\frac{E_{km}p_k(\mathbf{w}_{jmn}\mid\Theta_k)}{\sum_{k'}E_{k'm}p_{k'}(\mathbf{w}_{jmn}\mid\Theta_{k'})}\right]\\ &\quad\times\frac{\frac{\partial}{\partial\Theta_k}[E_{km}p_k(\mathbf{w}_{jmn}\mid\Theta_k)]}{E_{km}p_k(\mathbf{w}_{jmn}\mid\Theta_k)}\\ &= \sum_{j,m,n} p_0(\mathbf{w}_{jmn})P(k\mid jmn)\frac{\partial}{\partial\Theta_k}\ln[E_{km}p_k(\mathbf{w}_{jmn}\mid\Theta_k)].\end{aligned} \tag{27}$$

Here, we made use of the well-known identity $\partial\ln y(x)/\partial x = [1/y(x)]\partial y(x)/\partial x$, and the definition of

$$P(k\mid jmn) = \frac{E_{km}p_k(\mathbf{w}_{jmn}\mid\Theta_k)}{\sum_{k'}E_{k'm}p_{k'}(\mathbf{w}_{jmn}\mid\Theta_{k'})}. \tag{28}$$

Using (27), the set of parameters $\Theta_k$ that maximize $LL$ will satisfy

$$\frac{\partial LL}{\partial\Theta_k} = \sum_{j,m,n} p_0(\mathbf{w}_{jmn})P(k\mid jmn)\frac{\partial}{\partial\Theta_k}\ln p_k(\mathbf{w}_{jmn}\mid\Theta_k) = 0,\qquad k = 1,2,\ldots,K. \tag{29}$$

The maximization of $LL$ must also be performed with respect to the mixture weights $E_{km}$. However it is now necessary to incorporate the constraint from (10) using the method of Lagrange multipliers. Thus we seek to minimize the Lagrangian

$$F = -LL + \lambda\left(\sum_{k'}E_{k'm} - \sum_n p_0(\mathbf{w}_{jmn})\right) \tag{30}$$

with respect to $E_{km}$, where $\lambda$ is the Lagrange multiplier. Setting the partial derivative of $F$ to zero, i.e.,

$$\frac{\partial F}{\partial E_{km}} = -\frac{\partial LL}{\partial E_{km}} + \lambda = 0 \tag{31}$$

the following value for $\lambda$ is obtained

$$\lambda = \frac{\partial LL}{\partial E_{km}}. \tag{32}$$

An expression for $\partial LL/\partial E_{km}$ is now needed, and its derivation is similar to (27). We obtain

$$\begin{aligned}\frac{\partial LL}{\partial E_{km}} &= \sum_{j,n} p_0(\mathbf{w}_{jmn})P(k\mid jmn)\frac{\partial}{\partial E_{km}}\ln[E_{km}p_k(\mathbf{w}_{jmn}\mid\Theta_k)]\\ &= \frac{1}{E_{km}}\sum_{j,n} p_0(\mathbf{w}_{jmn})P(k\mid jmn).\end{aligned}$$

Combining this with (32),

$$\lambda E_{km} = \sum_{j=1}^{J}\sum_{n=1}^{N} p_0(\mathbf{w}_{jmn})P(k\mid jmn). \tag{33}$$

Next we perform the summation over $k$ on both sides of (33), then use (10) and (16) to simplify. This results in the solution for the Lagrange multiplier $\lambda = J$, where $J$ is the number of scan frames (time indices) from each sensor used in the estimation. Substituting this value back into (33), we obtain

$$E_{km} = \frac{1}{J}\sum_{j,n} p_0(\mathbf{w}_{jmn})P(k\mid jmn). \tag{34}$$

Equation (34) is satisfied by the set of mixture weights $E_{km}$ that maximize $LL$, subject to the constraint in (10).

A set of equations has now been derived governing the complete set of model parameters $\Theta = \{E_{km},\Theta_k\}$, for $k = 1,2,\ldots,K$ and $m = 1,2,\ldots,M$. Equation (29) governs the parameters $\Theta_k$, while (34) governs the mixture weights $E_{km}$. Unfortunately, solving for the parameters directly from these equations can be problematic. The difficulty arises because $P(k\mid jmn)$ is a function of all of the unknown parameters $\{E_{km},\Theta_k\}$ (see (28)). Thus, if there are $Q$ unknown parameters, it would be required to solve a coupled set of $Q$ nonlinear equations. However, the problem is not hopeless, and the maximum of $LL$ can be reached in an iterative fashion, alternating between updating $P(k\mid jmn)$, then updating the set of parameters $\{E_{km},\Theta_k\}$. We use the notation $P^{(I)}(k\mid jmn)$ and $\{E_{km}^{(I)},\Theta_k^{(I)}\}$ to indicate the values of these quantities at the $I$th iteration. Then, the explicit update formula for $P^{(I)}(k\mid jmn)$ is given by (15), while the parameters

are updated in order to satisfy modified versions of (29) and (34), i.e.,

$$E_{km}^{(I+1)} = \frac{1}{J}\sum_{j,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \tag{35}$$

and

$$\sum_{j,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \left[\frac{\partial}{\partial \Theta_k} \ln p_k(\mathbf{w}_{jmn} \mid \Theta_k)\right]_{\Theta_k = \Theta_k^{(I+1)}} = 0. \tag{36}$$

This iterative procedure is guaranteed to converge to a local maximum of $LL$, as shown in Appendix B. It will be of later use to note that

$$\sum_k E_{km}^{(I)} = \sum_n p_0(\mathbf{w}_{jmn}) \tag{37}$$

for all iterations $I$. This follows from (35) by using the fact that $\sum_k P^{(I)}(k \mid jmn) = 1$, as can be shown from (15).

The update formula in (35) for the mixture weight parameter is a simple, closed-form expression. The precise form of the update equation for $\Theta_k$ depends upon the particular model $p_k(\mathbf{w}_{jmn} \mid \Theta_k)$ that gets plugged into (36). In some cases, this leads to closed-form expressions, for example when estimating the covariance and mean parameters of Gaussian mixtures. Also, in this paper closed-form equations are obtained when estimating $\sigma_{dk}^2$ using (13) or when estimating the single-sensor range-only tracking parameters using (25). However, sometimes the model is specified such that a closed-form expression is not possible, for example when estimating the track parameters using multi-sensor data, as described at the end of Section IV. In this case, as an alternative to the closed-form update formula of (36), the parameters $\Theta_k$ can be updated by moving along the uphill gradient of $LL$, i.e.,

$$\begin{aligned}\Theta_k^{(I+1)} &= \Theta_k^{(I)} + h \cdot \left[\frac{\partial LL}{\partial \Theta_k}\right]_{\Theta_k = \Theta_k^{(I)}} \\ &= \Theta_k^{(I)} + h \cdot \sum_{j,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \\ &\quad \times \left[\frac{\partial}{\partial \Theta_k} \ln p_k(\mathbf{w}_{jmn} \mid \Theta_k)\right]_{\Theta_k = \Theta_k^{(I)}}.\end{aligned} \tag{38}$$

Here $h$ is the stepsize which can be chosen, for example, by trial and error. Note that the procedure described by (38) is not pure gradient ascent, since this step is alternated with updating $P^{(I)}(k \mid jmn)$ via (15) and $E_{km}^{(I)}$ using (35). In fact, this procedure is a special case of GEM [11]. Note that in some cases the set of parameters $\Theta_k$ may be split, with some members being updated using gradient ascent (38) and some being updated using a closed-form equation (36).

The general procedure, developed above, is valid for an arbitrary mixture model. These results are now used to derive the specific parameter estimation equations for the tracking application considered in this paper.

*Variance Parameters*: The recursive update formula is now derived for the variance parameter $\sigma_{dk}^2$. In order to use the update formula given by (36), it is necessary to compute an explicit formula for the factor within the square brackets, by substituting the specific model $p_k(\mathbf{w}_{jmn} \mid \Theta_k)$ used for the tracking application. Both the target model of (5) and the clutter model of (7) give the following result (neglecting irrelevant terms):

$$\begin{aligned}&\frac{\partial}{\partial (1/\sigma_{dk}^2)} \ln p_k(\mathbf{w}_{jmn} \mid \Theta_k) \\ &\quad = \frac{\partial}{\partial (1/\sigma_{dk}^2)} \left[\frac{1}{2}\ln(1/\sigma_{dk}^2) - \frac{1}{2}(1/\sigma_{dk}^2)(d_{jnm} - D_{jkm})^2\right] \\ &\quad = \frac{1}{2}\sigma_{dk}^2 - \frac{1}{2}(d_{jnm} - D_{jkm})^2.\end{aligned}$$

Note, it is actually more convenient to work with the inverse variance $1/\sigma_{dk}^2$ rather than the variance itself, as shown in the above equation. Substituting this expression into (36) leads to the update formula for the variance parameter described by (13). The update formula in (14) for the range variance $\sigma_{rk}^2$ is derived in a similar fashion.

*Track Parameters, Multi-Sensor Case*: Due to the model complexity, a closed-form update formula cannot be derived for the tracking parameters in the multi-sensor scenario. However, a GEM update formula can be derived, starting with (38), and computing the factor in square brackets by inserting the track model given by (5). Using $s_k$ as a generic placeholder for any of the track parameters $\{x_k^0, x_k', x_k'', y_k^0, y_k', y_k''\}$, we compute (neglecting irrelevant terms)

$$\begin{aligned}&\frac{\partial}{\partial s_k} \ln p_k(\mathbf{w}_{jmn} \mid \Theta_k) \\ &\quad = \frac{\partial}{\partial s_k}\left[-\frac{1}{2\sigma_{rk}^2}(r_{jnm} - R_{jkm})^2 - \frac{1}{2\sigma_{dk}^2}(d_{jnm} - D_{jkm})^2\right] \\ &\quad = \left(\frac{r_{jnm} - R_{jkm}}{\sigma_{rk}^2}\right)\left(\frac{\partial R_{jkm}}{\partial s_k}\right) + \left(\frac{d_{jnm} - D_{jkm}}{\sigma_{dk}^2}\right)\left(\frac{\partial D_{jkm}}{\partial s_k}\right).\end{aligned}$$

Inserting this expression into (38), the parameter update formula described by (21) and (22) is obtained.

*Track Parameters, Single Sensor Case*: The problem of range-only tracking using a single sensor is discussed in Section V. The goal is to find values for the track parameters $\{R_k^0, R_k', R_k''\}$, for each target component $k = 1,2,3,\ldots,(K-2)$, which maximize $LL$. In this case the target model given by (5) is still valid. However the equations relating $\{R_k^0, R_k', R_k''\}$ to the quantities $R_{jkm}$ and $D_{jkm}$ are now given by (23) and (24), respectively. Although there is only a

single sensor $m = 1$, the $m$ subscript is maintained in the quantities $R_{jkm}$ and $D_{jkm}$ for consistency with the expression for the target model in (5).

The update formulas for $\{R_k^0, R_k', R_k''\}$, are derived by starting with the general formula of (36), and substituting an explicit formula for the factor in square brackets, which is computed using the target model $p_k(\mathbf{w}_{jmn} \mid \Theta_k)$ in (5). Using $s_k$ as a generic placeholder for any of the parameters $\{R_k^0, R_k', R_k''\}$, the square-bracketed factor in (36) is

$$\begin{aligned}\frac{\partial}{\partial s_k}\ln p_k(\mathbf{w}_{jmn} \mid \Theta_k) &= \frac{\partial}{\partial s_k}\left[-\frac{1}{2}(1/\sigma_{rk}^2)(r_{jnm} - R_{jkm})^2 \right.\\ &\qquad \left. -\frac{1}{2}(1/\sigma_{dk}^2)(d_{jnm} - D_{jkm})^2\right]\\ &= \left(\frac{r_{jnm} - R_{jkm}}{\sigma_{rk}^2}\right)\left(\frac{\partial R_{jkm}}{\partial s_k}\right)\\ &\quad + \left(\frac{d_{jnm} - D_{jkm}}{\sigma_{dk}^2}\right)\left(\frac{\partial D_{jkm}}{\partial s_k}\right).\end{aligned}$$

Substituting this expression into (36), the parameter update formula

$$\left\langle\left[\left(\frac{r_{jnm} - R_{jkm}}{\sigma_{rk}^2}\right)\left(\frac{\partial R_{jkm}}{\partial s_k}\right) + \left(\frac{d_{jnm} - D_{jkm}}{\sigma_{dk}^2}\right)\left(\frac{\partial D_{jkm}}{\partial s_k}\right)\right]_{s_k = s_k^{(I+1)}}\right\rangle^{(I)} = 0 \tag{39}$$

is obtained where, for compactness, we have used the bracket notation $\langle * \rangle^{(I)}$ defined in (17). Note that (23) and (24) are used to express $R_{jkm}$ and $D_{jkm}$ and their derivatives with respect to $s_k = \{R_k^0, R_k', R_k''\}$ so, for example, $\partial R_{jkm}/\partial R_k^0 = 1$ and $\partial R_{jkm}/\partial R_k' = t_j$. For each target component $k$, (39) is used to produce three different equations for the three track parameters $s_k = \{R_k^0, R_k', R_k''\}$. First, substituting $s_k = R_k^0$ into (39), we obtain

$$(R_k^0)^{(I+1)}\langle 1\rangle^{(I)} + (R_k')^{(I+1)}\langle t_j\rangle^{(I)} + (R_k'')^{(I+1)}\langle t_j^2\rangle^{(I)} = \langle r_{jnm}\rangle^{(I)}. \tag{40}$$

Here, for example, $(R_k^0)^{(I+1)}$ denotes the updated estimate for $R_k^0$ computed at the $(I+1)$th iteration. Note that (40) is linear with respect to the three parameters $R_k^0$, $R_k'$, and $R_k''$. By substituting $s_k = R_k'$, then $s_k = R_k''$, into (39) we obtain two additional equations which are also linear with respect to $R_k^0$, $R_k'$, and $R_k''$. Thus there are three equations that are linear with respect to three unknowns which, together, can be expressed using matrix notation as shown in (25) given in Section V. Note there is a separate $3 \times 3$ set of equations for each target component $k = 1, 2, \ldots, (K-2)$. In each iteration the tracking parameters are updated by inverting these matrix equations.

## APPENDIX B. CONVERGENCE PROOF

This convergence proof generally follows the one given in [22, ch. 5.6.3], although here we provide some details that were left out in the reference. The proof is valid for the general mixture model, and is not restricted to the tracking model used in this paper.

In order to show convergence to a local maximum of $LL$, we show that $LL$ never decreases from one iteration to the next, i.e., $LL(\Theta^{(I+1)}) - LL(\Theta^{(I)}) \geq 0$, where $LL(\Theta^{(I)})$ is the log-likelihood computed in the $I$th iteration. From (11),

$$\begin{aligned}&LL(\Theta^{(I+1)}) - LL(\Theta^{(I)})\\ &\quad = \sum_{j,m,n} p_0(\mathbf{w}_{jmn})[\ln p(\mathbf{w}_{jmn} \mid \Theta^{(I+1)}) - \ln p(\mathbf{w}_{jmn} \mid \Theta^{(I)})].\end{aligned} \tag{41}$$

It is easy to see from (15) that $\sum_k P^{(I)}(k \mid jmn) = 1$, and therefore we can insert this unity factor into the above expression, i.e.,

$$\begin{aligned}&LL(\Theta^{(I+1)}) - LL(\Theta^{(I)})\\ &\quad = \sum_{j,m,n} p_0(\mathbf{w}_{jmn}) \sum_k P^{(I)}(k \mid jmn)\\ &\qquad \times [\ln p(\mathbf{w}_{jmn} \mid \Theta^{(I+1)}) - \ln p(\mathbf{w}_{jmn} \mid \Theta^{(I)})].\end{aligned} \tag{42}$$

We can further expand the expression by noting

$$\ln p(\mathbf{w}_{jmn} \mid \Theta^{(I)}) = \ln(E_{km}^{(I)} p_k(\mathbf{w}_{jmn} \mid \Theta_k^{(I)})) - \ln P^{(I)}(k \mid jmn)$$

as can be seen from the definition of $P^{(I)}(k \mid jmn)$ in (15). Using this fact, and rearranging terms, we can obtain

$$\begin{aligned}&LL(\Theta^{(I+1)}) - LL(\Theta^{(I)})\\ &\quad = \sum_{j,m,n} p_0(\mathbf{w}_{jmn}) \sum_k P^{(I)}(k \mid jmn) \ln\left[\frac{P^{(I)}(k \mid jmn)}{P^{(I+1)}(k \mid jmn)}\right]\\ &\qquad + \sum_{j,k,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \ln(E_{km}^{(I+1)} p_k(\mathbf{w}_{jmn} \mid \Theta_k^{(I+1)}))\\ &\qquad - \sum_{j,k,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \ln(E_{km}^{(I)} p_k(\mathbf{w}_{jmn} \mid \Theta_k^{(I)})).\end{aligned} \tag{43}$$

Since $\sum_k P^{(I)}(k \mid jmn) = 1$ and $P^{(I)}(k \mid jmn) \geq 0$ for all iterations $I$, the log-sum inequality [41] or Jensen's inequality can be used to prove the first term on the right-hand side of (43) is $\geq 0$. Thus, in order to show convergence we need to show the sum of the second and third terms is $\geq 0$, which can be done by analyzing the rules governing the updates of the parameters $\{E_{km}, \Theta_k\}$. The rule governing the update of the weights $E_{km}$ is given by (35), which can be rewritten as

$$\sum_{j,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \frac{1}{E_{km}^{(I+1)}} - J = 0. \tag{44}$$

Then, since

$$\frac{1}{E_{km}^{(I+1)}} = \left[\frac{\partial}{\partial E_{km}} \ln(E_{km} p_k(\mathbf{w}_{jmn} \mid \Theta_k))\right]_{E_{km}=E_{km}^{(I+1)}}$$

therefore (44) can be rewritten as[3]

$$\left[\frac{\partial}{\partial E_{km}} \left\{ \sum_{j,m',n} p_0(\mathbf{w}_{jm'n}) P^{(I)}(k \mid jm'n) \times \ln(E_{km'} p_k(\mathbf{w}_{jm'n} \mid \Theta_k)) - J E_{km} \right\}\right]_{E_{km}=E_{km}^{(I+1)}} = 0. \tag{45}$$

The rule governing the update of $\Theta_k$ is given by (36), which can be rewritten as

$$\left[\frac{\partial}{\partial \Theta_k} \left\{ \sum_{j,m',n} p_0(\mathbf{w}_{jm'n}) P^{(I)}(k \mid jm'n) \times \ln(E_{km'} p_k(\mathbf{w}_{jm'n} \mid \Theta_k)) - J E_{km} \right\}\right]_{\Theta_k=\Theta_k^{(I+1)}} = 0. \tag{46}$$

Observe that (45) and (46) contain the same expression within the curly brackets, which is a function of $E_{km}$ and $\Theta_k$ for indices $k = 1,2,\ldots,K$ and $m = 1,2,\ldots,M$. Furthermore, (45) and (46) imply the function within curly brackets has an extremum at the point $(E_{km} = E_{km}^{(I+1)}, \Theta_k = \Theta_k^{(I+1)})$. By computing the second derivatives of the curly bracketed function with respect to $E_{km}$ and $\Theta_k$ (not shown here), it is straightforward to show that the point $(E_{km} = E_{km}^{(I+1)}, \Theta_k = \Theta_k^{(I+1)})$ represents a global maximum, given the Gaussian components used in our model, and given that $P^{(I)}$ has been fixed using parameter values from the previous iteration. Thus,

$$\left[\sum_{j,m',n} p_0(\mathbf{w}_{jm'n}) P^{(I)}(k \mid jm'n) \times \ln(E_{km'}^{(I+1)} p_k(\mathbf{w}_{jm'n} \mid \Theta_k^{(I+1)})) - J E_{km}^{(I+1)}\right] \geq \left[\sum_{j,m',n} p_0(\mathbf{w}_{jm'n}) P^{(I)}(k \mid jm'n) \times \ln(E_{km'}^{(I)} p_k(\mathbf{w}_{jm'n} \mid \Theta_k^{(I)})) - J E_{km}^{(I)}\right]. \tag{47}$$

[3]It is important to recall that $P^{(I)}(k \mid jmn)$ is fixed using the parameter values $E_{km}^{(I)}$ and $\Theta_k^{(I)}$ which were computed in the previous $I$th iteration (refer to (15)). Therefore $P^{(I)}(k \mid jmn)$ is not a function of the variables $E_{km}$ and $\Theta_k$ which we seek to optimize in iteration $I + 1$.

It should be noted that if GEM is used to update $\Theta_k$ according to (38), then the inequality in (47) is still true (given a small enough stepsize) since (38) implies gradient ascent of the function in curly brackets of (46). Equation (47) can be modified by summing both sides over $k$, using (37) to simplify, then cancelling terms. The result is

$$\sum_{j,k,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \ln(E_{km}^{(I+1)} p_k(\mathbf{w}_{jmn} \mid \Theta_k^{(I+1)})) \geq \sum_{j,k,m,n} p_0(\mathbf{w}_{jmn}) P^{(I)}(k \mid jmn) \ln(E_{km}^{(I)} p_k(\mathbf{w}_{jmn} \mid \Theta_k^{(I)})).$$

Thus, the sum of the second and third terms on the right-hand side of (43) is $\geq 0$, which means $LL(\Theta^{(I+1)}) - LL(\Theta^{(I)}) \geq 0$. This implies convergence to a local maximum of $LL$.

## ACKNOWLEDGMENT

This research was supported by Dr. John Sjogren of the Air Force Office of Scientific Research. The authors would like to thank Jen King Jao and Jason Franz from Lincoln Laboratory for supplying the experimental radar data and for technical assistance.

## REFERENCES

[1] Reid, D.
An algorithm for tracking multiple targets.
*IEEE Transactions on Automatic Control*, **AC-24** (1979), 843–854.

[2] Blackman, S.
Multiple hypothesis tracking for multiple target tracking.
*IEEE Aerospace and Electronic Systems Magazine*, **19**, 1, pt. 2 (2004), 5–17.

[3] Fortmann, T., Bar-Shalom, Y., and Scheffe, M.
Sonar tracking of multiple targets using joint probabilistic data association.
*IEEE Journal of Oceanic Engineering*, **OE-8** (1983), 173–183.

[4] Bar-Shalom, Y., Kirubarajan, T., and Lin, X.
Probabilistic data association techniques for target tracking with applications to sonar, radar and EO sensors.
*IEEE Aerospace and Electronic Systems Magazine*, **20**, 8, pt. 2 (2005), 37–55.

[5] Bar-Shalom, Y., and Li, X. R.
*Multitarget-Multisensor Tracking: Principles and Techniques*.
Storrs, CT: YBS Publishing, 1995.

[6] Li, X., Wong, K. M., and Bosse, E.
An interior point linear programming approach to two-scan data association.
*IEEE Transactions on Aerospace and Electronic Systems*, **35** (1999), 474–490.

[7] Hue, C., Le Cadre, J. P., and Perez, P.
Tracking multiple objects with particle filters.
*IEEE Transactions on Aerospace and Electronic Systems*, **38** (2002), 791–812.

[8] Hue, C., Le Cadre, J. P., and Perez, P.
Sequential Monte Carlo methods for multiple target tracking and data fusion.
*IEEE Transactions on Signal Processing*, **50** (2002), 309–326.

[9] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T.
A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking.
*IEEE Transactions on Signal Processing*, **50** (2002), 174–188.

[10] Dempster, A. P., Laird, N. M., and Rubin, D. B.
Maximum likelihood from incomplete data via the EM algorithm.
*Journal of Royal Statistical Society*, Series B, **39** (1977), 1–38.

[11] McLachlan, G. J., and Krishnan, T.
*The EM Algorithm and Extensions*.
New York: Wiley, 1997.

[12] Redner, R. A., and Walker, H. F.
Mixture densities, maximum likelihood and the EM algorithm.
*SIAM Review*, **26** (1984), 195–239.

[13] Streit, R. L., and Luginbuhl, T. E.
Maximum likelihood method for probabilistic multi-hypothesis tracking.
*Proceedings of SPIE International Symposium, Signal and Data Processing of Small Targets 1994*, vol. 2335-24, Orlando, FL, Apr. 5–7, 1994.

[14] Streit, R. L., and Luginbuhl, T. E.
Probabilistic multi-hypothesis tracking.
NUWC-NPT technical report 10428, Naval Undersea Warfare Center, Newport, RI, Feb. 1995.

[15] Willett, P., Ruan, Y., and Streit, R.
PMHT: Problems and some solutions.
*IEEE Transactions on Aerospace and Electronic Systems*, **38** (2002), 738–754.

[16] Rago, C., Willett, P., and Streit, R.
Direct data fusion using the PMHT.
*Proceedings of the 1995 American Control Conference*, vol. 3, Seattle, WA, June 1995, 1698–1702.

[17] Kreig, M., and Gray, D.
Multi-sensor probabilistic multi-hypothesis tracking using dissimilar sensors.
*Proceedings of SPIE Conference 3086*, Apr. 1997, 129–138.

[18] Molnar, K. J., and Modestino, J. W.
Application of the EM algorithm for the multitarget/multisensor tracking problem.
*IEEE Transactions on Signal Processing*, **46** (1998), 115–129.

[19] Pulford, G. W., and La Scala, B. F.
MAP estimation of target manoeuver sequence with the expectation-maximization algorithm.
*IEEE Transactions on Aerospace and Electronic Systems*, **38** (2002), 367–377.

[20] Avitzour, D.
A maximum likelihood approach to data association.
*IEEE Transactions on Aerospace and Electronic Systems*, **28** (1992), 560–565.

[21] Perlovsky, L. I.
Fuzzy tracking of multiple objects.
*Proceedings of SPIE, Signal Processing, Sensor Fusion, and Target Recognition*, vol. 1699, Apr. 21, 1992, 18–20.

[22] Perlovsky, L. I.
*Neural Networks and Intellect*.
New York: Oxford University Press, 2001, ch. 7 and 9.

[23] Perlovsky, L. I.
Cramer-Rao bound for tracking in clutter and tracking multiple objects.
*Pattern Recognition Letters*, **18** (1997), 283–288.

[24] Perlovsky, L. I., Kwauk, R., and Tye, D.
MLANS application to classification while tracking.
*Proceedings of SPIE, Signal Processing, Sensor Fusion, and Target Recognition III*, vol. 2232, Apr. 4, 1994, 105–110.

[25] Perlovsky, L. I., Schoendorf, W. H., Tye, D. M., and Chang, W.
Concurrent classification and tracking using maximum likelihood adaptive neural system.
*Journal of Underwater Acoustics*, **45** (1995), 399–414.

[26] Perlovsky, L. I., Schoendorf, W. H., Garvin, L. C., Chang, W., and Monti, J.
Development of concurrent classification and tracking.
*Journal of Underwater Acoustics*, **47** (1997), 202–210.

[27] Ruan, Y., and Willett, P.
Multiple model PMHT and its application to the second benchmark radar tracking problem.
*IEEE Transactions on Aerospace and Electronic Systems*, **40** (2004), 1337–1347.

[28] Dunham, D. T., Dempster, R. J., and Blackman, S. S.
Tracking algorithm speed comparisons between MHT and PMHT.
*Proceedings of the Fifth International Conference on Information Fusion*, vol. 2, July 8–11, 2002, 846–851.

[29] Yanhua, R., Willett, P., and Streit, R.
Comparison of PMHT and S-D assignment trackers.
*IEE Colloquium on Target Tracking: Algorithms and Applications*, Nov. 11–12 1999, 12/1–12/4.

[30] Deming, R. W., Schindler, J., and Perlovsky, L. I.
Track-before-detect of multiple slowly moving targets.
*Proceedings of IEEE Radar Conference 2007*, Boston, MA, Apr. 17–20, 2007.

[31] Deming, R. W., Schindler, J., and Perlovsky, L. I.
Concurrent tracking and detection of slowly moving targets using dynamic logic.
*IEEE International Conference on Integration of Knowledge Intensive Multi-Agent Systems* (KIMAS '07), Waltham, MA, Apr. 29–May 3, 2007.

[32] Duda, R. O., and Hart, P. E.
*Pattern Classification and Scene Analysis*.
New York: Wiley, 1973, sect. 6.4.

[33] Stimson, G. W.
*Introduction to Airborne Radar*.
Mendham, NJ: Scitech Publishing, 1998, ch. 13, 165–167.

[34] Roberts, S. J., Husmeier, D., Rezek, I., and Penny, W.
Bayesian approaches to Gaussian mixture modeling.
*IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20** (1998), 1133–1142.

[35] Constantinopoulis, C.
Bayesian feature and model selection for Gaussian mixture models.
*IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28** (2006), 1013–1018.

[36] Figueiredo, M. A. T., and Jain, A. K.
Unsupervised learning of finite mixture models.
*IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24** (2002), 381–396.

[37] Perlovsky, L. I., Plum, C. P., Franchi, P. R., Tichovolsky, E. J., Choi, D. S., and Weijers, B.
Einsteinian neural network for spectrum estimation.
*Neural Networks*, **10** (1997), 1541–1546.

[38] Shore, J. E.
Minimum cross-entropy spectral analysis.
*IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-29** (1981), 230–237.

[39] Kullback, S.
*Information Theory and Statistics*.
New York: Wiley, 1959.

[40] Titterington, D.
On the iterative image space reconstruction algorithm for ECT.
*IEEE Transactions on Medical Imaging*, **MI-6** (1987), 52–56.

**Ross Deming** (M'04—SM'07) received the B.S. degree from Cornell University, Ithaca, NY, in 1985, the M.S. degree from the University of Vermont, Burlington, in 1993, and the Ph.D. degree from Northeastern University, Boston, MA, in 1996, all in electrical engineering.

Currently, he is a consultant for General Dynamics, Inc., where he develops radar and optical signal processing tools for the Air Force Research Laboratory. He was previously employed as an electrical engineer at General Electric, Inc. and Teradyne, Inc., and as a senior scientist at Nichols Research Corp. and Witten Technologies, Inc. While at Witten Technologies he was part of a small team that won, among other awards, the Wall Street Journal's 2004 Technology Innovation Award (1st place software category, honorable mention overall) for its technology that creates detailed images of objects and conditions underground using radar. He is also coauthor on a patent for this radar system.

**John K. Schindler** (S'62—M'64—SM'89—LF'91) received the SB degree from the Massachusetts Institute of Technology, Cambridge, and the M.S. and Ph.D. degrees from Purdue University, Lafayette, all in electrical engineering.

Prior to his retirement, he was the Director of Electromagnetics and Reliability at the Air Force Rome Laboratory where he directed research and development activities in electromagnetics, solid state electronics and reliability. In addition, he served on a defense task force to organize and manage the DoD electronics program. He is a lecturer in electrical engineering at Northeastern University, teaching graduate courses in radar, digital communications and statistical and adaptive signal processing. He has consulted on radar target classification, antenna design and radar cross section measurements. He has been an Adjunct Professor of Electrical Engineering at the Air Force Institute of Technology.

Dr. Schindler is a recipient of the IEEE Millennium Medal, the Meritorious Civilian Service Award and the Presidential Rank Award of Meritorious Executive.

**Leonid Perlovsky** (M'86—SM'95) is principal research physicist and technical advisor at the Air Force Research Laboratory. Previously, from 1985 to 1999, he served as chief scientist at Nichols Research, a $0.5 B high-tech organization, leading the corporate research in intelligent systems, neural networks, information science, sensor fusion, and algorithm development. He served as professor at Novosibirsk University and New York University. He participated as a principal in commercial startups developing tools for text understanding, biotechnology, and financial predictions. He published more than 250 papers, delivered invited keynote plenary talks, tutorial lectures, and authored a book *Neural Networks and Intellect*, (Oxford University Press, 2001 (currently in the 3rd printing)).

Dr. Perlovsky organizes conferences on computational intelligence, serves on multiple IEEE committees, chairs IEEE Boston Computational Intelligence Chapter, and received the Distinguished Member of the IEEE Boston Section Award. He serves as editor-at-large for *Natural Computations* journal and as editor-in-chief for *Physics of Life Reviews*. He received the IEEE Distinguished Member of Boston Section Award 2005; the USAF Research Lab Charles Ryan Memorial Award for Basic Research 2007; and International Neural Network Society Gabor Award for achievement in engineering/applications 2007.